Ontological modelling of geographical relationships for map generalization

Dutton, Geoffrey ; Edwardes, Alistair

Abstract: Understanding and encoding the roles that map features play in landscapes can assist making maps for particular purposes and for generalizing those maps appropriately. This paper explores how enumerating and classifying toponyms (place names) can enhance cartographic databases to allow applications to consider contextual relationships when choosing what to render or not and what symbolization rules to apply. A case study is presented of a coastal region in the U.S. to illustrate how ontologies can be formalized to support map display and generalization at a conceptual level that goes beyond traditional feature coding standards. While this might seem to some a return to a "capes and bays" geography (rote itemization of isolated geographic facts), it is in fact an effort to integrate toponyms into formalized knowledge framework expressed in a standardized, content- neutral language that can be shared by software anywhere on the World Wide Web.

Ontological Modeling of Geographical Relationships for Map Generalization

Geoffrey Dutton[1]* and Alistair Edwardes[2]

[1] 20 Payson Road, Belmont MA 02478 USA
+1-617-489-4524
gdutton@joimail.com

[2] Department of Geography, University of Zurich
Winterthurerstrasse 190, 8057 Zurich, Switzerland
aje@geo.unizh.ch

KEYWORDS: map generalization, semantic relationships, structural relationships, ontologies, spatial modeling, knowledge representation, OWL, Web Ontology Language, RDF, topographic maps, hydrographic charts

*Corresponding author

Abstract

Understanding and encoding the roles that map features play in landscapes can assist making maps for particular purposes and for generalizing those maps appropriately. This paper explores how enumerating and classifying toponyms (place names) can enhance cartographic databases to allow applications to consider contextual relationships when choosing what to render or not and what symbolization rules to apply. A case study is presented of a coastal region in the U.S. to illustrate how ontologies can be formalized to support map display and generalization at a conceptual level that goes beyond traditional feature coding standards. While this might seem to some a return to a "capes and bays" geography (rote itemization of isolated geographic facts), it is in fact an effort to integrate toponyms into formalized knowledge framework expressed in a standardized, content-neutral language that can be shared by software anywhere on the World Wide Web.

Introduction

Digital cartographic data almost always fail to include and also lack access to information describing the roles of the geographic features they represent in the landscapes they inhabit. This impedes developing specialized applications from general-purpose geodata, and makes generalizing maps much harder than it need be. Problems are especially acute when preparing maps of the same data for different purposes or audiences. Without accounting for the *roles* of features in a landscape and in a map, it is difficult to select, simplify, displace and resymbolize features appropriately.

One way to augment cartographic and other geodata is to identify features and portions of them as subclasses of an entity type, as is commonly done to designate roads as divided or undivided highways, local roads, fire roads, etc. This approach is often termed *database enrichment*; it involves adding information about names, attributes, and topological relationships. Enrichment can suffice when the database is always used to make the same type of map at a limited range of scales. However, when constructing maps for alternative purposes, the enrichments may not suffice to determine how to generalize the data. It is quite a different exercise, for example, to compile a navigational chart or a tourist map from data designed to produce topographic quadrangles.

We believe that using *ontologies* can help solve such problems by describing what data items, their properties, and relationships mean in operational ways. In particular, we see a need to connect the "worldview" of an application (i.e.. the purpose and audience of a map) to the schema of the (hopefully enriched) database containing geodata to be rendered as a map.

"Ontology" is more than a philosophical term and a trendy tech buzzword; it is a necessary prerequisite for modeling and processing data to inform complex applications, in this instance digital cartography. As a proper noun, Ontology is the most general branch of metaphysics, concerned with the nature of being. Computationally and in common usage, an ontology constitutes a systematic inventory of what is known about a domain of knowledge.

We explore developing, formalizing, and linking formal ontologies for geographic phenomena and vector cartographic representations of them. We are particularly concerned with structural and semantic properties and relationships of geographic phenomena and how these can be cartographically depicted and generalized when the roles of features in a map can vary. For example, a small coastal promontory may take on heightened significance for a marine chart if it defines a cove used as a harbor, has offshore rocks, or is occupied by a lighthouse or radio beacon.

After cataloging some semantic and structural relationships and after describing how these relationships have been exploited to derive and enforce generalization constraints, we focus on how differing geographical ontologies for waterbodies and coastal features can affect cartographic interpretations. We illustrate how ontological modeling of a geographic, real-world object can help to portray of it in accordance with map scale and purpose.

To investigate the formal modeling of ontologies we use the syntax of the Web Ontology Language (OWL). The aim of OWL is to allow applications to interpret and integrate diverse information available on the Web. Although the current application is not aimed at or limited to Web environments, OWL's expressive, neutral and standardized approach to modeling semantics make it well-suited for its purposes. In addition, OWL provides possibilities to incorporate the developed ontologies with continuing research on generalization web services through a common XML foundation.

OWL is an XML-based language for defining and sharing ontologies. It builds on the Resource Description Framework (RDF) as well as earlier web ontology standards DAML+OIL. RDF presents a set of simple semantics for modeling data in XML. It allows the definition of class hierarchies and properties of classes. Properties permit predicates to be specified that relate classes to other classes or to data types, enabling subject-predicate-object type statements to be made different kinds of entities. OWL extends RDF by introducing richer methods for representing the semantics of entities and their properties. It allows, for example, classes to be defined as disjoint from or equivalent to other classes, and allows restrictions on the values and cardinality of properties. Logical characteristics of properties can also be specified (e.g. transitive or symmetric); these enable automated reasoners to infer logical entailments between predicates and thereby support the representation of semantic knowledge.

OWL comes in three flavors; *OWL Lite*, *OWL DL* and *OWL Full*. *OWL Lite* provides the simplest and least complex subset. It is best suited for describing taxonomies and thesauri with simple constraints. *OWL DL* is based on description logic. This maximizes the expressiveness of the language while ensuring logical inference remains computationally tractable. *OWL Full* is the most expressive and the least strict; it does not ensure that inferences using it will come to completion. For this project, we used OWL DL.

Context of Generalization

Researchers have long recognized the necessity of considering context when generalizing map features, and have been grappling with some of these hard issues. *Context* has been perceived in a number of ways, some of which are listed below from the more concrete interpretations to the more abstract. When contextually generalizing map features, it may be important to

1. Discover and convey how portions of them differ in character
2. Assess their intrinsic importance at presentation scale
3. Collapse or combine them in order to maintain their presence
4. Communicate their status to their neighbors
5. Adapt certain ones to changes in their neighbors
6. Consider the roles that they have in landscapes and maps
7. Switch context according to the purpose of the map

The closer to the top of this list that an activity is, the more that data structures for geometry, topology, and feature class and attributes can inform it. The closer to the bottom it is, the more dependence an activity has on knowledge of the real world in various domains. Because the scope such global knowledge often transcends feature data, it is difficult to encode in geographic data structures. Because it is so often linguistic and qualitative in nature, such knowledge is difficult to formalize in ways useful to digital cartography. Operational connections between semantics and spatial structure are hard to make.

Operationalizing Cartographic Context

Consider the differences between topographic maps and nautical charts, which display many of the same features in coastal zones. It is not necessary to conduct a study to identify many of them. When decreasing scale, topographic maps

1. Simplify shapes of coastlines and clusters of islands
2. Depict geomorphologic features fairly uniformly

3. Delete roads of lower classes or that dead-end
4. Remove solitary buildings and structures

On the other hand, when nautical charts decrease scale, they

1. Simplify coastlines and islands to preserve their importance to mariners
2. Depict only those geomorphologic features visible from offshore
3. Preserve certain main roads and some that access a shoreline
4. Depict built structures that might be useful to navigation

How many land-based data models describe what a harbor consists of, or can help to preserve its identity at decreased scale? How many marine-based data models describe the character of built-up land or can help to preserve its role at smaller scales?

Spatial Semantic Relationship Representation Issues

Considerable progress has been made in discovery of topological and proximal relationships. For instance, Jones, Bundy & Ware (1995) use a constrained Delaunay triangulation; Gold (1994). recommends using line Voronoi model. While clearly useful and powerful, such structures may burden databases with large amounts of information that may never be used, but still need to be updated when changes occur; on-the-fly structuring (Jones, Kidner & Ware, 1994; Ruas, 1995) is one way to limit demands on databases, but at the expense of run-time performance. Regardless of whether representation is static or dynamic, Delaunay and Voronoi methods cannot identify all structural relationships, and further processing is always needed to identify relationships that are germane to a particular application. Furthermore, having identified existing semantic and spatial relationships in a neighborhood, most of them will probably never be salient for generalizing features there. As a rule, generalization is only invoked when there is a problem (e.g., features would become too small to be displayed or too cluttered). Hence, many relationships are more usefully computed ad hoc and on demand, rather than being represented explicitly throughout a persistent database.

Semantic properties and relations needed for generalization that cannot be inferred from geometry must be coded explicitly. To enable this, the database environment must support appropriate representation mechanisms, such as

1. Hierarchical feature representation that builds arbitrarily complex features from geometric primitives and their encoded attributes
2. The possibility of sharing geometric primitives between features, and
3. Ways to maintain attributes that describe relationships between features as well as features themselves.

As few existing systems perform such data modeling, only a minority of the relationships described in the examples of the previous sections can be easily represented using unenhanced commercial GIS platforms.

For those spatial and semantic relationships that are discoverable analytically, decisions must be made about when to do this and what to do with the results of such analyses. As both discovery and representation of object relationships entail considerable difficulties, it remains an open question whether semantic and structural information is best represented explicitly (i.e. stored in a database) or implicitly (i.e. inferred on demand by computation), as Ruas & Lagrange (1995) discusses. Here, we simply point out that explicit representation tends to become less useful:

- the more features are involved in a relationship;
- the more 'transient' a relationship is (i.e. it only has meaning for a specific purpose or scale);
- the more requirements there are for complex auxiliary data structures that maintain parallel 'shadow' representations (e.g., triangulations, Voronoi diagrams, strip trees);
- the less effort is required to discover relevant relationships from scratch, and
- the less suitable representation mechanisms available in the target system are for encoding given semantic properties

As it is difficult to specify geographic ontologies that are not application-specific, attempts to design a general-purpose spatial database and populate it with semantics supportive of a range of applications are not likely to be very successful. Thus we recommend starting with a limited application scope and build outwards to encompass additional feature classes and application ontologies, one at a time.

Feature Relationships and Roles

Modeling of semantics of spatial data has also received much attention lately in the context of GIS interoperability. Schema integration, for instance, necessitates the assessment of semantic similarities between feature class (Bishr, 1998); (Devogele, Trevisan & Raynal, 1996). While schema integration is restricted to semantic properties, the actual data integration process must further analyze semantic and geometric (structural) relationships inherent in the datasets to be merged; (Devogele, Trevisan & Raynal, 1996; Uitermark, van Oosterom, Mars & Molenaar, 1998; Sester, Anders & Walter, 1998). Schema compatibility issues arise naturally in object-oriented data modeling (Coad & Yourdon, 1991; Rumbaugh, Jacobson & Booch, 1999), as such methods give database architects wide latitude in how phenomena are represented.

In order to generalize a feature-oriented spatial database, holistic solutions are needed as well as object-specific ones. Constraints (limitations to solutions) need to be defined, and can be represented as objects or embedded as object properties or methods. Many constraints are in fact simple parameters coupled with representation rules (e.g., minimum areas, widths, and separation distances, line weights and symbol diameters) controlling representation of classes of features. Other constraints, potentially useful in generalization, are more complex, can be difficult to quantify and are not easily represented in an object-oriented context. These involve relationships among features and provide semantic information about their behavior and roles. For example, certain features such as buildings and access roads, parks and monuments, bridges and rivers, etc., tend to have logical relationships, such that one should not be eliminated or moved without regard to another. It is useful to analyze map data to identify interdependent features, and then link their semantics to allow such objects to point to one another.

Information about map features useful in making generalization decisions can be drawn from outside sources (e.g., attribute files or metadata) or user knowledge, a process generally de-scribed as database enrichment (Uitermark, van Oosterom, Mars & Molenaar, 1998). Conflating contextual data from outside sources is not a simple process in the general case, and even when successfully performed objects in the recipient database must be instructed how to use this information. This requires infusions of *knowledge* as well as of *data*. It is our belief that a great deal of knowledge takes the form of data that no one has yet learned how to use.

Current implementations of automated map generalization tend to ignore (hence may fail to conserve) various *roles* features have in a landscape and relationships they have with one another. Such knowledge can assume a variety of forms and levels of meaning. It can range from simple facts concerning individual features (such as a building being a historic monument, or a run of a river being a main branch or a tributary) to facts that relate pairs of features (such as indicating a road that leads to a campground or a lake which is the water supply for a town). Similarly, larger groups of features (such as orderings of networks such as rivers, highways and railroads or alignments of building footprints characterized by their spacing and orientations) also exhibit semantics. Some structural knowledge can be discovered using analytic tools such as minimum spanning trees and shape statistics that relate a set of lakes, or a Delaunay triangulation that itemizes proximities of a set of buildings and roads.

A given geometric pattern or topologic arrangement may or may not be cartographically appropriate, depending on the functions of features within a landscape, the landscape's ontological interpretation, and the map's scale and specific mission. One can attempt to deconstruct data models to infer what world-views they represent, in order to be able to map elements between them. We feel this is not a reliable exercise, as it is dangerous to infer world-views that have not been explicitly stated. Rather, we prefer to take account of world-views as early as possible when making models.

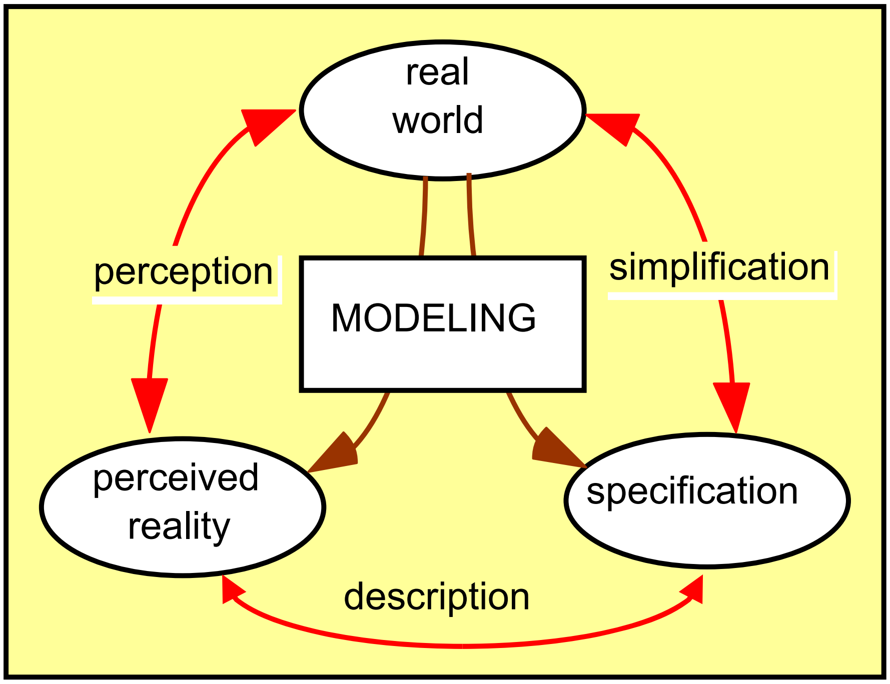

Figure 1: Aspects of Modeling Reality (adapted from Salgé, 1995)

Figure 1, from (Salgé 1995), illustrates the situation quite succinctly. It postulates a physical existence ("real world") that is perceptually modeled (as "perceived reality" or "nominal ground"; in French a "terrain nominal") by those who regard it, and can be formally modeled (as a "specification") for purposes of symbolic manipulation. There is a tendency to ignore the nominal ground and directly refer only to the real world when constructing or interpreting a model. This, we believe, is a source of inconsistency that impedes comparison (hence interoperability) of different schemata. All perceptions of the real world are selective and scale-bound, thus constitute abstractions. A formal specification or model is also an abstraction, and therefore an abstraction of an abstraction. Thus the path labeled description in figure 1 is quite critical, and we assert, all too often ignored in spatial data modeling. We explored this link in this study.

Comparing Topographic Maps and Hydrographic Charts

National mapping and other agencies create series of topographic maps and nautical charts. Even though they represent many of the same physical features, and may even share some source data, they have different compilation rules, symbologies and scales. The two types of maps tend to be used by different consumers and for different purposes. They also get generalized in different ways. Comparing such maps can illuminate how the ontologies underlying them are related (they have many similarities); it also can illustrate how merging the ontologies would provide a stronger basis for modeling hydrographic data for making and generalizing either type of map.

The View from the Land

The southeastern coast of Massachusetts contains both densely-populated areas and natural areas featuring salt marshes, dunes, headlands, bays, channels, and many islands. Tourism, recreation, boating and fishing are among its biggest industries.

One such area is Sippican Harbor, in the town of Marion. A composite of two 1:24K USGS topographic maps from different dates is shown below (later revision on left). Note the detail with which wetlands are delineated and the stippled representation of inter-tidal and beach areas. Also note that while streets and individual dwellings are depicted, few docks and breakwaters are shown.

Figure 2: Sippican Region Represented on a 1:24K Topographic Map

The View from the Water

The map below is a detail from the 1:40K Buzzards Bay NOAA navigational chart of the same area (MassGIS, 2001). The built-up area in the center of Marion is glossed by a dark brown tint; no structures are displayed except for buoys in the water and some towers and pipelines. Wetlands and marshes are depicted, but only tinted in gray, much less prominently than in the map above. Topographic contours and streamlines abruptly end as they move away from the harbor, although the hill on Great Hill Point is delineated and called out (along with the water tank atop it). The water areas contain more soundings, and some of the bathymetric contours are suppressed in favor of them.

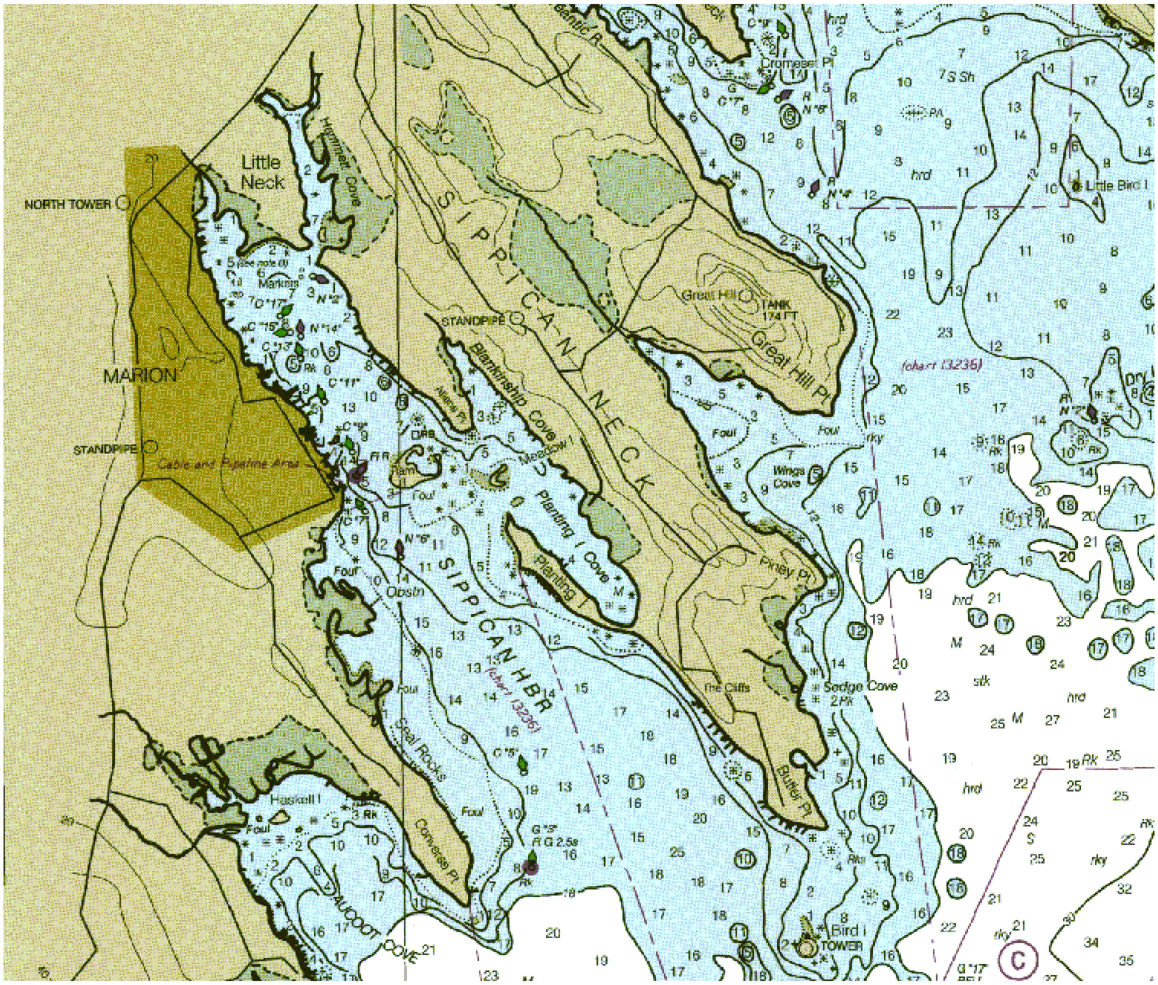

Figure 3: Sippican Region Represented on a 1:40K Nautical Chart

Other than for streets, the two maps share many toponyms. Notice that in both maps the names of proximal land and water features often come in pairs:

- Sippican Neck – Sippican Harbor
- Converse Neck – Aucoot Cove
- Planting Island – Planting Island Cove
- Great Hill Point – Wings Cove

The bays, harbors, and coves plus the peninsulas, necks, and points constitute a duality of topographic and bathymetric hierarchies that, when formalized and suitably encoded, can connect the view from the land and the view from the water. The following schematic map shows this; the red links and nodes show the land-based hierarchy, starting with southeastern Massachusetts, and the blue links and nodes show the water-based hierarchy, starting with Buzzards Bay:

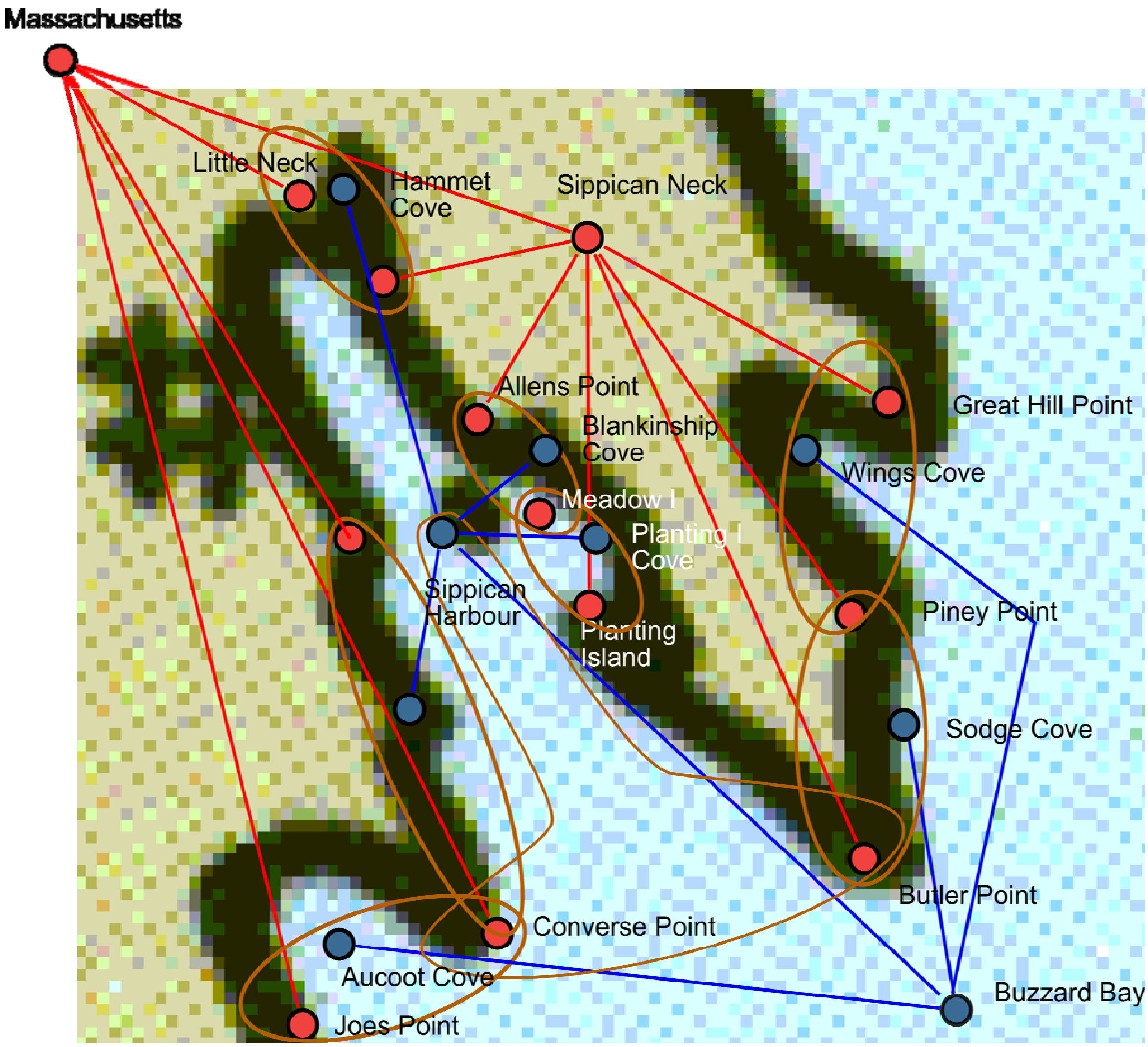

Figure 4: Toponymic Trees for Sippican Region Land and Water Features

The brown ovals have been inserted to indicate that blue nodes usually correspond to a pair of red nodes. Thus the two feature trees are intimately associated, even though the depths of corresponding nodes within them may not be the same. Once enumerated, connections between land and water nodes provide a way to associate the two databases on the basis of matched toponyms. Thus, when a generalization operator removes, for example, Great Hill Point or Piney Point, the application can understand that Wings Cove may no longer exist as a consequence. For this to happen, the generalization operator needs to report that it removed Great Hill and Piney points. While important, the means by which these identified features label line segments in their vicinity is beyond the scope of this paper.

An abstracted view of selected elements of the land-based subsumption hierarchy is shown below, starting with the entire state of Massachusetts:

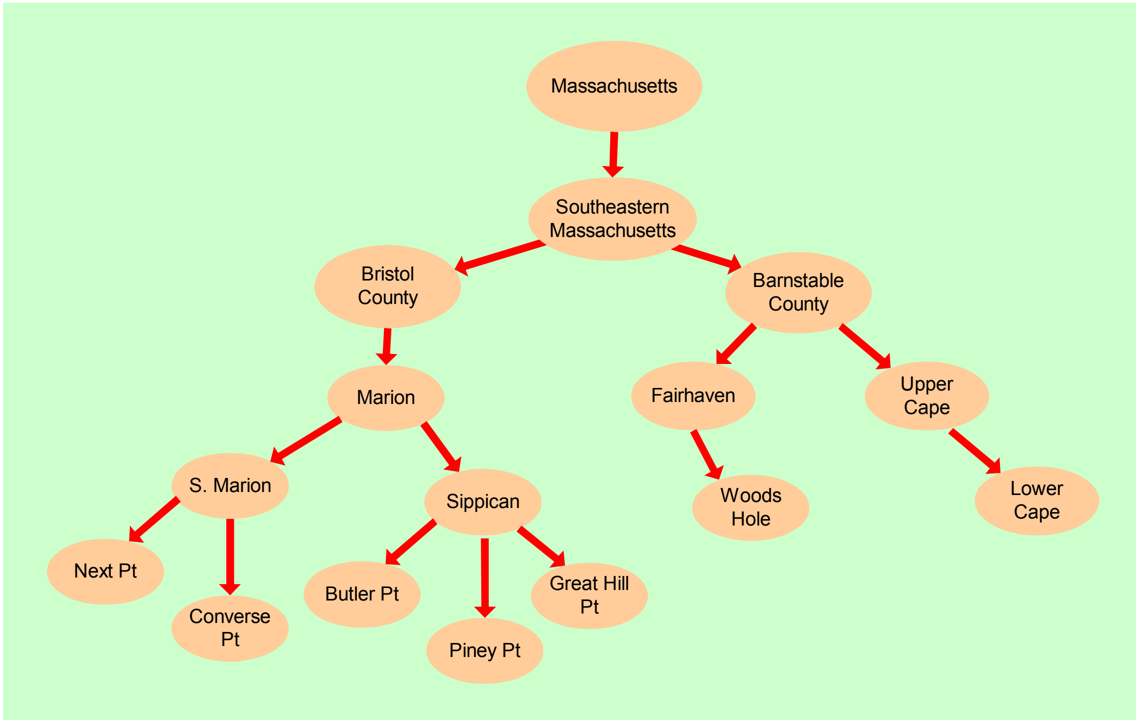

Figure 5: Toponymic Tree for Part of Massachusetts

Modeling Ontologies using Formal methods

From an engineering standpoint, much of the work on ontologies has been driven by the creation of languages allowing the semantics of concepts to be formally described. In part, the development of such frameworks has been driven by the need for independent domain models for software development and data description that are more open to reuse and exchange (especially over the Web) by making their semantics more explicit. Such formalisms allow this be integrating formal logics with object modeling. The Web Ontology Language (OWL) is one of the most recent of these using a dialect of formal logic called description logics (Baader *et al* 2003). The aim of OWL is to allow applications to interpret and integrate diverse information available online. Whilst the application described below is not limited to such environments, aspects of OWL as an expressive, neutral and standardised language for modeling semantics make it a well-suited candidate for these purposes.

Ontology Models and Object-Oriented Modeling

Modeling with OWL DL shares similarities with object oriented design. For example, each has a fundamental concept of a *class* which can have *properties* and *instances*. Both allow the creation of *generalization* and *specialization* type class hierarchies and allow individuals to be aggregated with part-whole type property relations. However, there are significant differences also. Classes in OWL are sets of individuals rather than types. This means that a class can be defined in terms of the individuals that make it up. Classes are assumed to overlap unless specifically defined as disjoint, meaning that just because an individual is asserted to be in one class doesn't mean that it cannot also belong to another. An important consequence of these factors for generalization is that both class membership and the creation and definition of classes can be change dynamically at runtime in response to generalization decisions. Properties in OWL are defined independently of classes, allowing subsumption relationships between classes sharing properties to be inferred. In addition, the semantics of properties is enriched through *characteristics*. For example, a property can be defined as functional (relates to at most one other individual), transitive (e.g. if A *hasProperty* B and B *hasProperty* C then A *hasProperty* C), or symmetric.

Modeling Classes in OWL

In OWL classes are modeled by asserting membership conditions. Such conditions are based on different kinds of relation. The *subsumption* relation ('is a') is used to assert class hierarchies by logical implication. Saying class B is a subsumed A is equivalent to saying that all individuals of class B are also individuals of class A. All classes are initially subsumed from the base class 'owl:Thing'. Other conditions can be asserted using unions and intersections of other classes or restrictions on properties. For example, a landform might be described as those individuals that are *madeOf* dry land.

The conditions that describe a class can be "necessary", or "necessary and sufficient". Necessary conditions describe a class in terms of the conditions which individual members of it must be satisfying. Classes that are only *described* with necessary conditions are termed primitive. Necessary and sufficient conditions define a class from the perspective of the individuals. Essentially if an individual or class is found to satisfy the set conditions it must be a member of that class also. For example, an island class might be defined as consisting landforms of dry land completely surrounded by water which are not continents. Such classes are said to be *defined*. Reasoners can be used to classify, individuals or classes subsumed by a defined class automatically.

Modelling a Coastal Region Ontology in OWL

A small ontology was developed for the Sippican Bay example to evaluate OWL. The OWL editor *Protégé* (Protégé, 2006) was used for this purpose together with the *FaCT++* reasoner (Fact++, 2006).

As with all modeling, there is no unique way of describing the phenomena under study. Particularly, with OWL there are many ways to say the same things. OWL is *open world*, in that things not stated are assumed to be true. Consequently it is very easy to say things that are not meant. We followed the approach of Rector (2003, 2004) to try to manage the modeling process through the application of *normalisation rules*.

Rector suggests modeling domain concepts as a number of disjoint, primitive trees; hierarchies that employ only single inheritance and only necessary conditions. He terms these the *primitive skeleton*. They consist of two main types of entities:

1. *Self-standing concepts* such as real world features, processes and roles
2. *Refining concepts* such as partitions of size (e.g., small, medium, large)

Classes to be inferred by a reasoner are then defined based on the classes of this skeleton.

For the example, four basic trees were modeled:

- *Structures* – geographic phenomena found in the maps
- *Roles* – different map purposes
- *Substances* – different physical substances
- *Modifiers* – refining concepts.

The figures expand the parts of the skeleton relevant to each of these trees. It should be noted that OWL itself is an XML format, but for the purposes of clarity the visualization capabilities of Protégé are used to present the ontology.

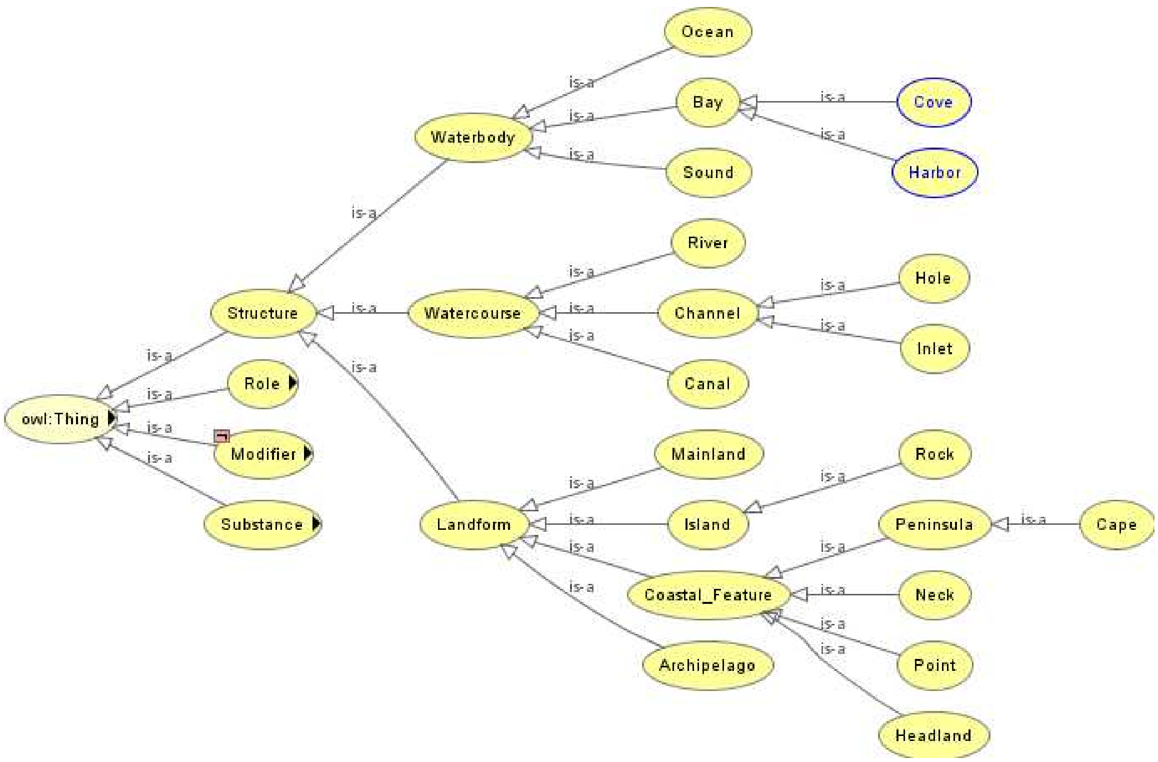

Figure 6: Structures in the Sippican Ontology

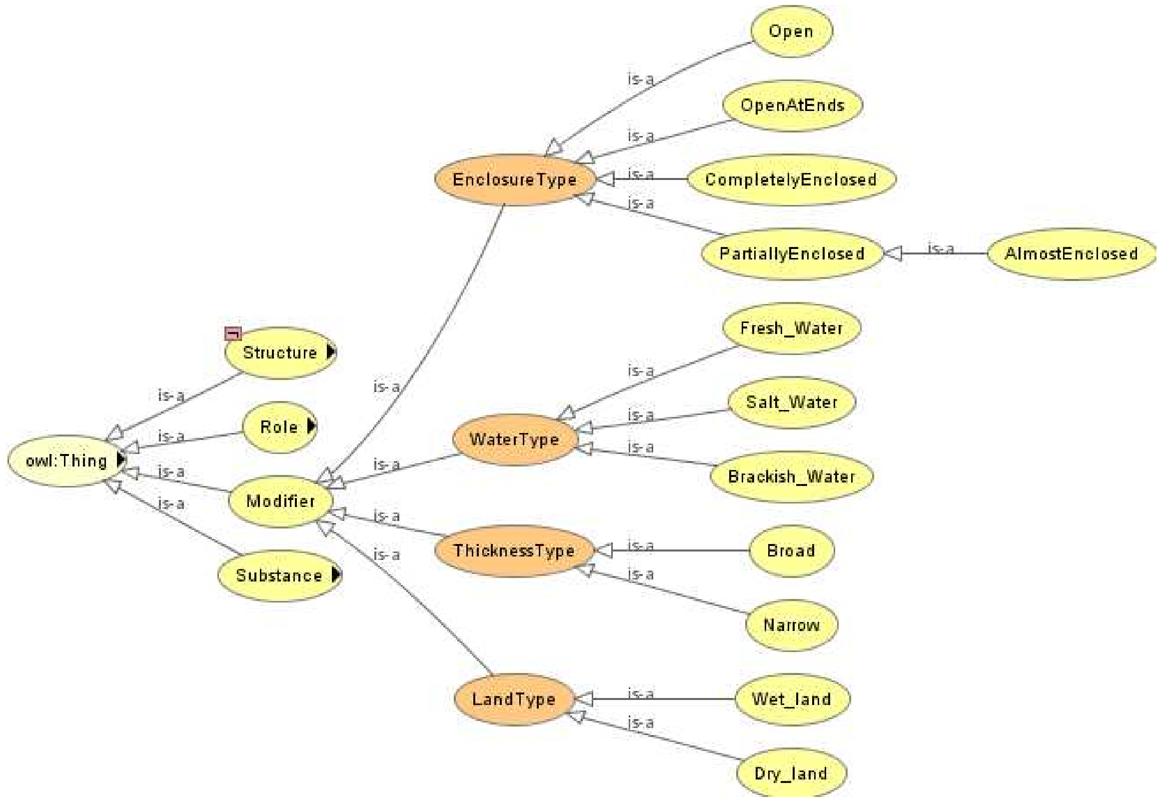

Figure 7: Modifiers in the Sippican Ontology

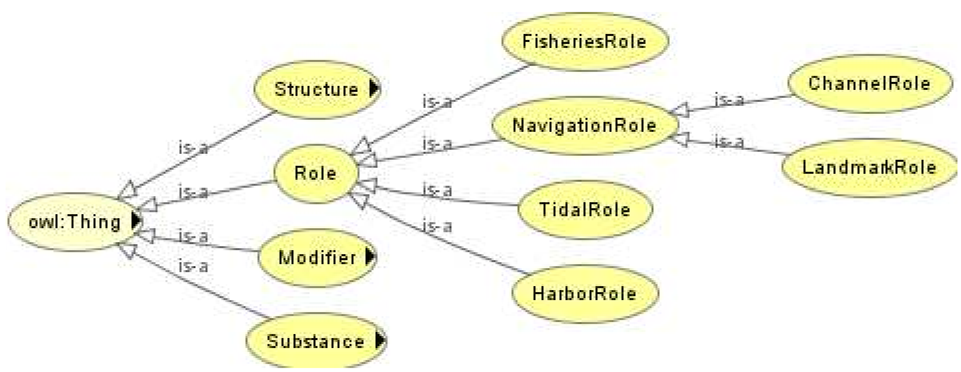

Figure 8: Roles in the Sippican Ontology

The next stage was to add individual relating to these classes. However, because the current generation of reasoners is poor at inferring relationships over individuals, individuals were instead also modeled as classes.

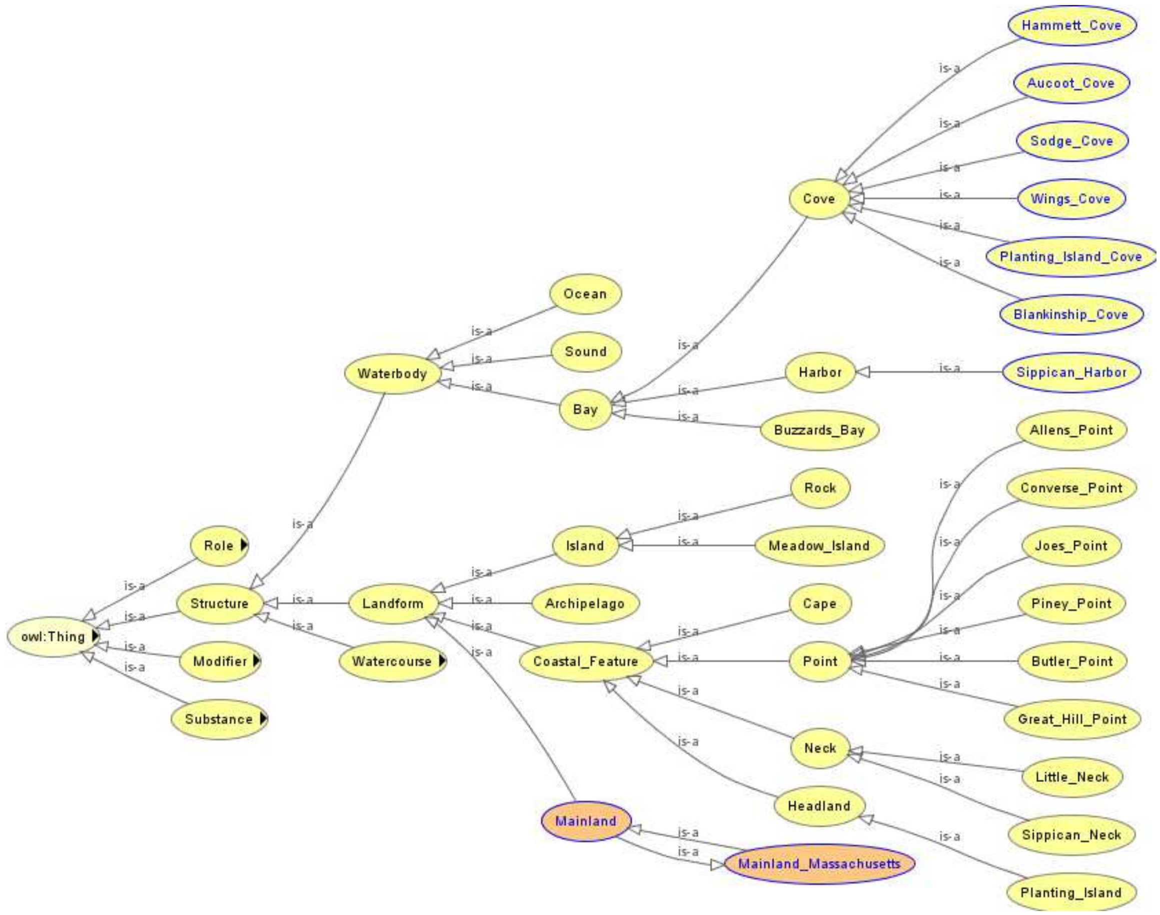

Figure 9: Instances in the Sippican Ontology

In the diagram, Mainland has a different colour because it was restricted to only include the subclasses of Mainland. Since these consist of only Mainland Massachusetts the relation therefore becomes reciprocal.

A number of properties where also added to the model. An example of such a property is a relation to model trees of features. This defined a transitive property for its parent and an inverse property for its children. For example Sippican Harbor was described with the necessary condition "$\exists$ parent Buzzards_Bay". This expresses that the class has (some values of) Buzzards Bay as a parent. A similar relation was defined for all children of Buzzards Bay and all children of Sippican Harbor. An example of inferring over this relation was through the creation of a class *Buzzard_Bay_Feature*s, which was defined as consisting of those classes with a parent of Buzzards Bay i.e. $\exists$ parent Buzzards_Bay. The figure below shows the action of classifying the ontology with the FaCT++ reasoner. Note that multiple inheritance is inferred within this hierarchy, though the asserted hierarchy is still single inheritance.

Figure 10: Reasoned Inferences about Buzzards Bay Features

A second transitive property *partOf* was used to relate waterbodies to landforms, for example, to relate Planting Island to Planting Island Cove. In addition, a property *hasRole* was defined that allowed different roles of classes to be described. For example 'points' were restricted to the *LandmarkRole* class and coves to the *HarborRole* class. An example of a class reasoning over these relationships is *Sippican_Navigation_Features*, which comprises features that are *partOf* Sippican Harbor, have a parent of Sippican Harbor or are *partOf* a class that has Sippican Harbor as a parent. In addition, it is restricted to the classes that have navigation or harbor roles

The following statements describe these constraints logically:

(∃ partOf Sippican_Harbor) ⊔ (∃ parent Sippican_Harbor) ⊔ (∃ partOf (∃ parent Sippican_Harbor))
∃ hasRole (NavigationRole ⊔ HarborRole)

The figure shows the result of classifying the class with the reasoner.

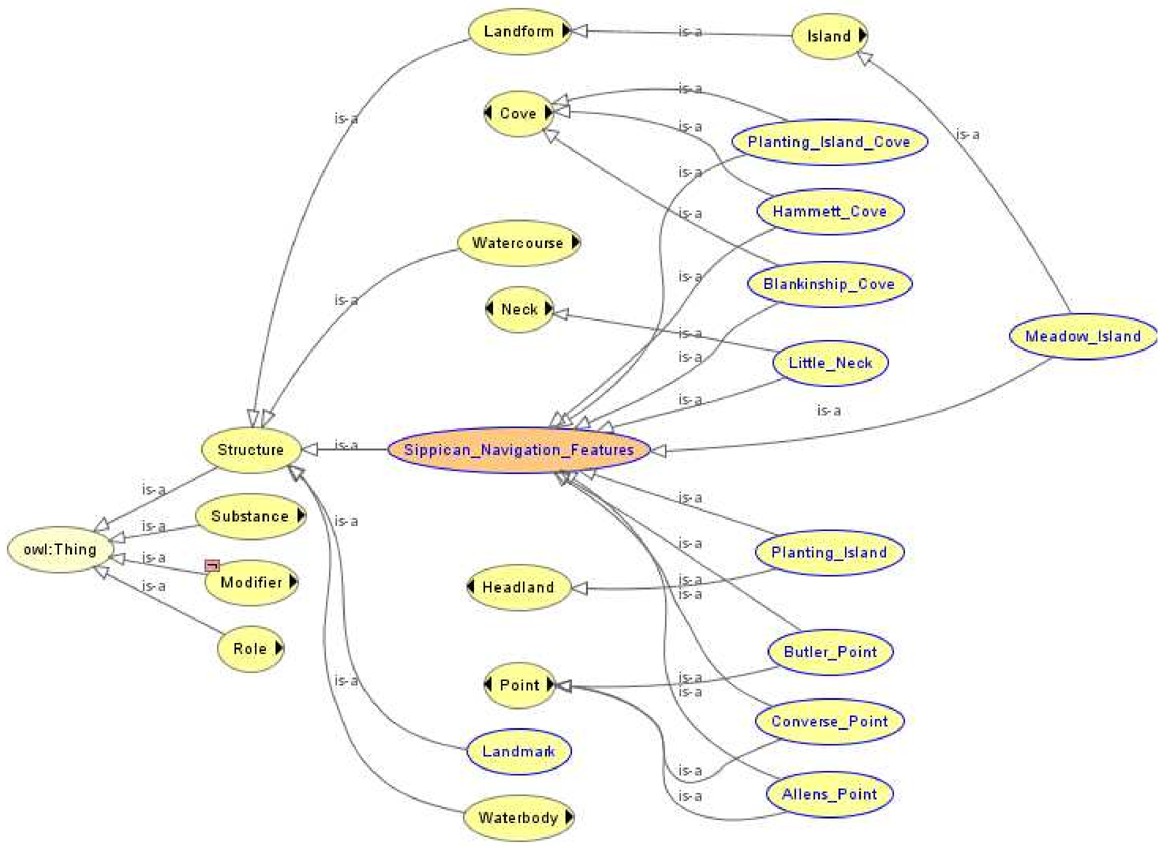

Figure 11: Sippican Bay Features in the Sippican Ontology

Conclusions and Outlook

Even though the ontology we modeled is far from complete, it serves to show how domain knowledge – such as the concepts of bay, cape, and headland – can be modeled and instantiated with specific examples for an area of interest. Such a domain model can be created independently and re-used for other examples, which could provide an important input for the task of database enrichment. In addition it demonstrates how structural relationships can be described relatively easily using semantics and how an application ontology (e.g. marine navigation) based on roles can be integrated to support the generalization of specific types of maps.

Developing the ontology impressed us with the importance of the value of local, specific knowledge, which we gathered from our study of maps of the region and from prior experience. For example, maps taught us that Planting Island is not an island in the conventional sense but actually a headland. Such knowledge is often overlooked in work both on ontologies and map generalization, but is an area where national mapping agencies could add value to their cartographic and toponymic data products. More complete models of semantic relations could also serve to determine where gaps in knowledge about features exist in a cartographic database.

Effective use of ontologies in cartographic production faces many serious challenges:

- Ontologies are painstaking to develop
- Ontologies for different applications will take on different forms
- Linking ontologies to cartographic data has yet to be automated
- Incorporating ontologies in GIS processing will be resource-intensive

Performing this study impressed us of the value of attending to toponymy (cataloging place names and attaching concepts to them), both for creating formal ontologies and for achieving deeper understanding of particular regions. We found ourselves identifying patterns that help us assemble a view of what was unique and thus significant, as well as representative of a pattern (and thus significant), in the manner of assembling a jig-saw puzzle. Such insights enabled us to craft and refine ontologies, which while still inadequate, show promise of maturing should we continue to attend to the lay of the land and the significance of the sea. “Capes and bays” are worth learning about, after all.

Acknowledgements

Robert Weibel co-authored the unpublished paper on which this one is based. The authors are indebted to Frank Brazile for suggestions on the constraints and semantic relationships of hydrographic features. The initial research was supported in part by Esprit project AGENT (LTR 24939, BBW-No. 97.0005).

References

Abdelmoty, A.I., Paton, N.W., Williams, H. Fernandes, A., Barga, M.L. & Dinn, A. (1994). Geographic data handling in a deductive object-oriented database. in *Proc. 5th Int. Conf. On Databases and Expert Systems Applications* (DEXA), D. Karagiannis (ed.), (pp. 445-454). Springer,

Baader, F., Horrocks, I., and Sattler, U. (2003) Description Logics. Volume Handbook on Ontologies in Information Systems of International Handbooks on Information Systems, chapter I: Ontology Representation and Reasoning, pages 3-31. Steffen Staab and Rudi Studer, Eds., Springer..

Bishr, Y. (1998). Overcoming the Semantic and other Barriers to GIS Interoperability. *International Journal of Geographical Information Science*, 12(4), p.p 299-314.

Coad, P. & Yourdon, E. (1991). *Object Oriented Analysis*. Engelwood Cliffs NJ: Yourdon Press, 233 pgs.

Devogele, T., Trevisan, J., & Raynal, L. (1996). Building a Multi-Scale Database with Scale Transition Relationships. In Molenaar, M. and Kraak, M.-J. (eds.): *Advances in GIS II* (Proc. SDH '96), pp. 337-351. London: Taylor & Francis.

Fact++ (2006) Open source reasoner for OWL-DL, http://owl.man.ac.uk/factplusplus/
Frank, A.U. (1997). Spatial Ontology: A Geographical Point of View. In Stock, O. (ed.): *Spatial and Temporal Reasoning*. Dordrecht: Kluwer Academic Publishers,
Gold, C. (1994). Three Approaches to Automated Topology, and How Computational Geometry Helps. *Proc. 6th International Symposium on Spatial Data Handling* (SDH 94), Edinburgh,, pp. 145-158.
Gruber, T.R. (1994). Toward principles for the design of ontologies used for knowledge sharing. *Formal Ontology in Conceptual Analysis and Knowledge Representation* (N. Guarino and R. Poli, eds.). Kluwer,.
Jones, C.B., Bundy, G.Ll. & Ware, J.M. (1995). Map Generalization with a Triangulated Data Structure. *Cartography and Geographic Information Systems,* 22(4), pp. 317-331.
Jones, C.B., Kidner, D.B. & Ware, M. (1994). The implicit triangular irregular network and multiscale databases. *The Computer Journal*, 37(1), pp, 43-57.
MassGIS (2001). Datalayers/GIS Database, NOAA Nautical Chart Images; http://www.mass.gov/mgis/noaacharts.htm
Protégé (2006) Open Source Ontology Editor and Knowledge Base Framework. http://protege.stanford.edu/
Rector, A. (2003) Modularisation of domain ontologies implemented in description logics and related formalisms including OWL, Proceedings of K-CAP:(ed J Genari). ACM. pp 21-129
Rector, A. (2004) Owl pizzas: ...Common errors and common patterns in OWL. Proceedings of EKAW 2004 (eds Motta, Shadbolt, et al) LNCS LNAI3257 pp 63-81
Ruas, A. (1995). Multiple Paradigms for Automating Map Generalization: Geometry, Topology, Hierarchical Partitioning and Local Triangulation. *ACSM/ ASPRS Annual Convention and Exposition,* Vol. **4** (Proc. Auto-Carto 12), pp. 69-78.
Ruas, A. & Lagrange, J.-P. (1995). Data and Knowledge Modeling for Generalization. In: Muller, J-C., Lagrange, J.-P., and Weibel, R. (eds.): *GIS and Generalization: Methodological and Practical Issues*, pp. 73-90. London: Taylor & Francis.
Rumbaugh, J., Jacobson I, & Booch, G. (1999). *Unified Modeling Language Reference Manual*. Reading MA: Addison-Wesley Longman, 550 pgs.
Salgé, F. (1995). Semantic accuracy. *Elements of Spatial Data Quality*, In S.C. Guptill and J.L. Morrison (eds.), pp. 139-151. Pergamon.
Sester, M., Anders, K.-H., & Walter, V. (1998). Linking Objects of Different Spatial Data Sets by Integration and Aggregation. *GeoInformatica*, **2**(4), pp. 335-358.
Smith, B. & Mark, D.M. (1998). Ontology and geographic kinds. *Proc. 8th Int. Symp. On Spatial Data Handling* (SDH98), Vancouver, July 1998, pp. 308-418.
Tang, A.Y., Adams, T.M., & Usery, E.L. (1996). A spatial data model for feature-based geographical information systems. *Int. J. of GIS* (10: 5), pp. 643-659.
Timpf, S. (1998). Map Cube Model – a model for multi-scale data. *Proc. 8th Int. Symp. On Spatial Data Handling* (SDH98), Vancouver, July 1998, pp. 190-201.
Uitermark, H., van Oosterom, P., Mars, N., & Molenaar, M. (1998). Propagating Updates: Finding Corresponding Objects in a Multi-source Environment.. *Proc. 8th Int. Symp. On Spatial Data Handling* (SDH98), Vancouver, July 1998, pp. 580-591.
Weibel, R. & Dutton, G. (1998). Constraint-based automated map generalization. *Proc. 8th Int. Symp. On Spatial Data Handling* (SDH98), Vancouver, July 1998, pp. 214-224.